IN THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

1. (**currently amended**): A graphics processor, comprising:

a plurality of parallellized graphics computational units; and

one or more task allocation units programmed to bypass defective ones of said [[subunits]] units within said groups, and to distribute incoming tasks only among operative ones of said [[subunits]] units.

2. (canceled)

3. (**currently amended**): The graphics processor of claim 1, wherein each of said parallellized graphics computational units [[include]] also includes respective multiple vertex processors.

4. (**currently amended**): The graphics processor of claim 1, wherein each of said parallellized graphics computational units [[include]] also includes respective texturing pipelines.

5. (**currently amended**): The graphics processor of claim 1, wherein each of said parallellized graphics computational units [[include]] also includes a respective memory [[controllers]] controller.

6. (canceled)

7. (original): A method of 3D graphics rendering which comprises: using a task allocation unit and parallellized graphics computational units with relations as recited in claim 1.

8. (**new**): The graphics processor of claim 1, wherein one or more of said parallellized graphics computational units operate with no more than 4 operative vertex processors.

9. (**new**): The graphics processor of claim 1, wherein one or more of said parallellized graphics computational units operate with no more than 4 operative texturing pipelines.

10. (**new**): The graphics processor of claim 4, wherein said texturing pipelines also include a shading unit and a texture filter unit.

11. (**new**): The graphics processor of claim 4, wherein said texturing pipelines also include a shading unit and a primary texture cache.

12. (**new**): A method of 3D graphics rendering, comprising the actions of:

providing a plurality of parallellized graphics computational units;

bypassing defective ones of said units, and

distributing incoming tasks only among operative ones of said units.

13. (**new**): The method of claim 12, wherein each of said parallellized graphics computational units also includes respective multiple vertex processors.

14. (**new**): The method of claim 12, wherein one or more of said parallellized graphics computational units operate with no more than 4 operative vertex processors.

15. (**new**): The method of claim 12, wherein each of said parallellized graphics computational units also includes respective multiple texturing pipelines.

16. (**new**): The method of claim 12, wherein one or more of said parallellized graphics computational units operate with no more than 4 operative texturing pipelines.

17. (**new**): The method of claim 12, wherein each of said parallellized graphics computational units also includes a respective memory controller.

18. (**new**): The method of claim 15, wherein said texturing pipelines also include a shading unit and a texture filter unit.

19. (**new**): The method of claim 15, wherein said texturing pipelines also include a shading unit and a primary texture cache.

20. (**new**): A computer graphics system comprising:

means for providing a plurality of parallellized graphics computational units;

means for bypassing defective ones of said units, and

means for distributing incoming tasks only among operative ones of said units.

21. (**new**): The system of claim 20, wherein each of said parallellized graphics computational units also includes respective multiple vertex processors.

22. (**new**): The system of claim 20, wherein one or more of said parallellized graphics computational units operate with no more than 4 operative vertex processors.

23. (**new**): The system of claim 20, wherein each of said parallellized graphics computational units also includes respective multiple texturing pipelines.

24. (**new**): The system of claim 20, wherein one or more of said parallellized graphics computational units operate with no more than 4 operative texturing pipelines.

25. (**new**): The system of claim 20, wherein each of said parallellized graphics computational units also includes a respective memory controller.

26. (**new**): The system of claim 23, wherein said texturing pipelines also include a shading unit and a texture filter unit.

27. (**new**): The system of claim 23, wherein said texturing pipelines also include a shading unit and a primary texture cache.

28. (**new**): A method for computer graphics system operation, comprising the actions of:

providing a plurality of parallellized rendering units;

bypassing defective ones of said units, and

distributing incoming tasks only among operative ones of said units.

29. (**new**): The method of claim 28, wherein each of said parallellized rendering units also includes respective multiple vertex processors.

30. (**new**): The method of claim 28, wherein one or more of said parallellized rendering units operate with no more than 4 operative vertex processors.

31. (**new**): The method of claim 28, wherein each of said parallellized rendering units also includes respective multiple texturing pipelines.

32. (**new**): The method of claim 28, wherein one or more of said parallellized rendering units operate with no more than 4 operative texturing pipelines.

33. (**new**): The method of claim 28, wherein each of said parallellized rendering units also includes a respective memory controller.

34. (**new**): The method of claim 31, wherein said texturing pipelines also include a shading unit and a texture filter unit.

35. (**new**): The method of claim 31, wherein said texturing pipelines also include a shading unit and a primary texture cache.